

FILEID**OUTCHA

N 7

000000 UU UU TTTTTTTTTT CCCCCCCCCC HH HH AAAAAA
000000 00 00 UU UU TTTTTTTTTT CCCCCCCCCC HH HH AAAAAA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
00 00 00 UU UU TT CC HH HH AA AA
000000 UUUUUUUUUU TT CCCCCCCCCC HH HH AAAAAA
000000 000000 UUUUUUUUUU TT CCCCCCCCCC HH HH AA AA

LL IIIII SSSSSSSS
LL IIIII SS SSSSSS
LLLLLLLLLL IIIII SSSSSSSS
LLLLLLLLLL IIIII SSSSSSSS

OUT
V04

```
1 0001 0 XTITLE 'Processor for chapter and appendix headers.'  
2 0002 0 MODULE outcha ( IDENT = 'V04-000'  
3 0003 0           XBLISS32[, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE  
4 0004 0           NONEXTERNAL = LONG_RELATIVE)]  
5 0005 0           ) =  
6 C006 1 BEGIN  
7 0007 1 :-----  
8 0008 1 *****  
9 0009 1 :  
10 0010 1 :* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
11 0011 1 :* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
12 0012 1 :* ALL RIGHTS RESERVED.  
13 0013 1 :*  
14 0014 1 :* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
15 0015 1 :* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
16 0016 1 :* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
17 0017 1 :* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
18 0018 1 :* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
19 0019 1 :* TRANSFERRED.  
20 0020 1 :*  
21 0021 1 :* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
22 0022 1 :* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
23 0023 1 :* CORPORATION.  
24 0024 1 :*  
25 0025 1 :* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
26 0026 1 :* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
27 0027 1 :*  
28 0028 1 :*  
29 0029 1 :-----  
30 0030 1 :  
31 0031 1 :  
32 0032 1 :++  
33 0033 1 : FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS  
34 0034 1 :  
35 0035 1 : ABSTRACT: Processor for chapter and appendix headers.  
36 0036 1 :  
37 0037 1 : ENVIRONMENT: Transportable  
38 0038 1 :  
39 0039 1 : AUTHOR: K.A. ALDEN      CREATION DATE: February, 1983  
40 0040 1 :
```

: 42 0)41 1 %SBTTL 'Revision History'
: 43 0)42 1 MODIFIED BY:
: 44 0043 1
: 45 0044 1 019 KFA00019 Ken Alden 14-Sep-1983
: 46 0045 1 Added emphasis support for autotitle if user used STCH or
: 47 0046 1 STAX for emphasis. This is for DSRPLUS only.
: 48 0047 1
: 49 0048 1 018 KFA00018 Ken Alden 23-Aug-1983
: 50 0049 1 Removed eight lines of code that are only needed for
: 51 0050 1 autosubtitle. These lines were left over from OUTHDR
: 52 0051 1 and prevented autotitle from carrying over any emphasis
: 53 0052 1 the user may have given with the chapter/appendix directives.
: 54 0053 1
: 55 0054 1 017 KFA00017 Ken Alden 28-Jul-1983
: 56 0055 1 Tweaked the flip bit from #16.
: 57 0056 1
: 58 0057 1 016 KFA00016 Ken Alden 27-Jul-1983
: 59 0058 1 Fixed logic for flip that was put in during #14.
: 60 0059 1
: 61 0060 1 015 KFA00015 Ken Alden 28-Jun-1983
: 62 0061 1 Made all calls to this module force a "chapter-oriented" book.
: 63 0062 1 Unnumbered chapters will now also reset the page number to one.
: 64 0063 1
: 65 0064 1 014 KAD00014 Keith Dawson 18-May-1983
: 66 0065 1 Removed call to PUTTPG for FLIP output.
: 67 0066 1
: 68 0067 1 013 KFA00013 Ken Alden 4-May-1983
: 69 0068 1 Un-numbered chapter/appendices now will not change the
: 70 0069 1 "chapter" status or page numbers of the document.
: 71 0070 1
: 72 0071 1 012 KFA00012 Ken Alden 2-May-1983
: 73 0072 1 Altered the way subtitles are cleared at the start of
: 74 0073 1 a chapter. The way it works now is: 1)AUTOTITLE is first
: 75 0074 1 checked. If true, then clear the subtitle. 2) If AUTOTITLE
: 76 0075 1 is false, the subtitle is clear anyway if AUTOSUBTITLE is true.
: 77 0076 1
: 78 0077 1 011 KFA00011 Ken Alden 29-Apr-1983
: 79 0078 1 Added code for the STARTODD bit;
: 80 0079 1
: 81 0080 1 010 REM00010 Ray Marshall April-1983
: 82 0081 1 Conditionalized for DSRPLUS so it can be used by DSR, too.
: 83 0082 1
: 84 0083 1 009 RER00009 Ron Randall 07-Apr-1983
: 85 0084 1 For DSRPLUS: Initialize footnote number.
: 86 0085 1
: 87 0086 1 008 KFA00008 Ken Alden 16-Mar-1983
: 88 0087 1 PUSH/POP_SCA now visible to DSR.
: 89 0088 1
: 90 0089 1 007 KFA00007 Ken Alden 9-Mar-1983
: 91 0090 1 TOC action was changed so the footnote MRA is always
: 92 0091 1 passed to the TOC as well as always being read. If
: 93 0092 1 ALTOTITLE is on, then the title MRA is also read. This
: 94 0093 1 passes the bolding/underlining information to the BRN, regardless.
: 95 0094 1
: 96 0095 1 006 KAD00006 Keith Dawson 07-Mar-1983
: 97 0096 1 Global edit of all modules. Updated module names, idents,
: 98 0097 1 copyright dates. Changed require files to BLISS library.

OUTCHA
V04-000

Processor for chapter and appendix headers.
Revision History

D 8
16-Sep-1984 01:19:03
14-Sep-1984 13:07:31

VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI;1

Page 3
(2)

: 99
: 100 0098 1 !--
 0099 1

OUT
V04

```
: 102      C100 1 %SBTTL 'Module Level Declarations'  
: 103      01C1 1  
: 104      0102 1 TABLE OF CONTENTS:  
: 105      0103 1  
: 106      0104 1 FORWARD ROUTINE  
: 107      0105 1 OUTCHA : NOVALUE;           ! Generate the CHAPTER or APPENDIX header  
: 108      0106 1  
: 109      0107 1 INCLUDE FILES:  
: 110      0108 1  
: 111      0109 1 LIBRARY 'NXPORT:XPORT';      ! XPORT Library  
: 112      0110 1 REQUIRE 'REQ:RNODEF';        ! RUNOFF variant definitions  
: 113      0241 1  
: 114      U 0242 1 %IF dsrplus %THEN  
: 115      U 0243 1 LIBRARY 'REQ:DPLLIB';       ! DSRPLUS BLISS Library  
: 116      0244 1 %ELSE  
: 117      0245 1 LIBRARY 'REQ:DSRLIB';        ! DSR BLISS Library  
: 118      0246 1 %FI  
: 119      0247 1  
: 120      0248 1  
: 121      0249 1 : EQUATED SYMBOLS:  
: 122      0250 1  
: 123      0251 1 EXTERNAL LITERAL  
: 124      0252 1 RINTES : UNSIGNED (8);  
: 125      0253 1  
: 126      0254 1 EXTERNAL LITERAL  
: 127      0255 1 S_FMRA;                  ! Allocated length for footnote MRA.  
: 128      0256 1  
: 129      0257 1 : OWN STORAGE:  
: 130      0258 1  
: 131      0259 1  
: 132      0260 1  
: 133      0261 1 : MACROS:  
: 134      0262 1  
: 135      0263 1 MACRO          ! NOTE: this code depends on the arrangement  
: 136      M 0264 1 show_counter =      ! of literals in TOCRTY.  
: 137      M 0265 1 ?  
: 138      M 0266 1   (.caption_minor_type MOD 3) EQL 2  
: 139      M 0267 1     OR  
: 140      M 0268 1     .caption_minor_type EQL min_chapt  
: 141      M 0269 1     OR  
: 142      M 0270 1     ;caption_minor_type EQL min_append  
: 143      M 0271 1 ;  
: 144      0272 1 %:  
: 145      0273 1  
: 146      0274 1 OWN          PP_SCA : $H_R_SCA_BLOCK;    !Used in PUSH_SCA, POP_SCA macros (defined in SCA.REQ).  
: 147      0275 1  
: 148      0276 1  
: 149      0277 1 : EXTERNAL REFERENCES:  
: 150      0278 1  
: 151      0279 1 EXTERNAL  
: 152      0280 1   ECC : SECC_BLOCKVECTOR,  
: 153      0281 1   FNCT : FNCT_DEFINITION,  
: 154      0282 1   FOMRA : FIXED STRING,  
: 155      0283 1   FOOTSF : VECTOR [TSF_SIZE],  
: 156      0284 1   FS01 : FIXED STRING,  
: 157      0285 1   GCA : GCA_DEFINITION,  
: 158      0286 1   HCT : HCT_DEFINITION,
```

```
; 159      0287 1   HLDSP : VECTOR [MAX_LEVELS],  
; 160      0288 1   HLLIST : COUNTED_LIST,  
; 161      0289 1   IRA : FIXED STRING,  
; 162      0290 1   MRA : REF FIXED STRING,  
; 163      0291 1   TITMRA : FIXED STRING,  
; 164      0292 1   TITTSF : VECTOR,  
; 165      0293 1   NPAGEN : PAGE DEFINITION,  
; 166      0294 1   PAGEN : PAGE DEFINITION,  
; 167      0295 1   PHAN : PHAN DEFINITION,  
; 168      0296 1   SBTMRA : FIXED STRING,  
; 169      0297 1   SBTTSF : VECTOR,  
; 170      0298 1   SCA : SCA_DEFINITION,  
; 171      0299 1   TSF : TSF_DEFINITION;  
; 172      0300 1  
; 173      0301 1 EXTERNAL  
; 174      0302 1   KHAR;  
; 175      0303 1  
; 176      0304 1 EXTERNAL ROUTINE  
; 177      0305 1   endchr,    endwrd,      gcpage,      gcskip,  
; 178      0306 1   gtpc,       guskip,      outcrg,      outctr,  
; 179      0307 1   outnj,     pacsec,      pacxxx,      putcnt,  
; 180      0308 1   puttpg,    putttx,      remmrg,      remneg,  
; 181      0309 1   rskips,    scant,       sdxy,       setcas,  
; 182      0310 1   titles,     tstblk;
```

```

: 185      0312 1 %SBTTL 'OUTCHA -- Process chapter & appendix headers'
: 186      0313 1 GLOBAL ROUTINE outcha
: 187      0314 1 (
: 188      0315 1     lines_before,          lines_after,          test_page_amount,
: 189      0316 1     counter_major_type,   counter_minor_type,
: 190      0317 1     counter_value,        counter_display_code,  counter_spaces_after,
: 191      0318 1     counter_pre_string_length, counter_pre_string_ptr,
: 192      0319 1     counter_post_string_length, counter_post_string_ptr,
: 193      0320 1     caption_major_type,    caption_minor_type,    caption_case,
: 194      0321 1     caption_is_centered,  caption_is_flush_right, caption_is_run_in,
: 195      0322 1     caption_is_bold,       caption_is_underlined, put_into_mem_file,
: 196      0323 1     autosubtitle,       brn_open,           break_before_caption,
: 197      0324 1     lines_between,       new_page,           startodd,
: 198      0325 1     tocpage
: 199      0326 1 ) : NOVALUE =
: 200      0327 1
: 201      0328 1 !++
: 202      0329 1 FUNCTIONAL DESCRIPTION:
: 203      0330 1 Processor for chapter and appendix headers.
: 204      0331 1
: 205      0332 1 FORMAL PARAMETERS:
: 206      0333 1
: 207      0334 1
: 208      0335 1 LINES BEFORE
: 209      0336 1 LINES BETWEEN
: 210      0337 1 LINES AFTER
: 211      0338 1 TEST PAGE AMOUNT
: 212      0339 1 BREAK BEFORE CAPTION
: 213      0340 1 COUNTER MAJOR TYPE
: 214      0341 1 COUNTER MINOR TYPE
: 215      0342 1 COUNTER VALUE
: 216      0343 1 COUNTER DISPLAY CODE
: 217      0344 1 COUNTER SPACES AFTER
: 218      0345 1 COUNTER_PRE_STRING_LENGTH
: 219      0346 1 COUNTER_PRE_STRING_PTR
: 220      0347 1 COUNTER_POST_STRING_LENGTH
: 221      0348 1 COUNTER_POST_STRING_PTR
: 222      0349 1 CAPTION MAJOR TYPE
: 223      0350 1 CAPTION MINOR TYPE
: 224      0351 1 CAPTION CASE
: 225      0352 1 CAPTION IS CENTERED
: 226      0353 1 CAPTION IS FLUSH RIGHT
: 227      0354 1 CAPTION IS RUN IN
: 228      0355 1 CAPTION IS BOLD
: 229      0356 1 CAPTION IS UNDERLINED
: 230      0357 1 NEW PAGE
: 231      0358 1 STARTODD
: 232      0359 1 TOCPAGE
: 233      0360 1 PUT INTO MEM FILE
: 234      0361 1 AUTOSUBTITLE
: 235      0362 1 BRN OPEN
: 236      0363 1
: 237      0364 1 IMPLICIT INPUTS: None
: 238      0365 1 IMPLICIT OUTPUTS: None
: 239      0366 1
: 240      0367 1
: 241      0368 1

```

leave this many blank lines before the header
 leave n blank lines between 'CHAPTER' or 'APPENDIX' & caption.
 leave this many blank lines after the (non-run-in) header
 start a new page unless this many lines remain
 start a new line between the counter and caption.
 (not implemented) always MAJ_RUNOFF
 specifies type of header: HL, Example, Figure, Table
 (not used for HLS) numerical value of the counter
 (not used for HLS) display code of the counter
 leave this many spaces between counter and caption
 (not used for HLS) length of pre-counter string
 (not used for HLS) pointer to pre-counter string
 (not used for HLS) length of post-counter string
 (not used for HLS) pointer to post-counter string
 (not implemented) always MAJ_RUNOFF
 specifies whether or not to show the counter
 code for the case rules to apply to the caption
 True if caption should be centered
 (not implemented) True if caption should be flush-right
 True if caption should be run-in with following text
 True if caption should be bolded
 True if caption should be underlined
 start a new page before CHAPTER or APPENDIX.
 Always TRUE for DSR.
 force the first page of a chapter to start on an odd page.
 insert a page number in the TOC.
 (not implemented) True if header should be output in .MEM file
 True if header should be picked up as an auto subtitle
 True if header should be sent to the TOC

```
: 242      0369 1 ! ROUTINE VALUE:  
.: 243      0370 1 ! COMPLETION CODES:    None  
.: 244      0371 1 ! SIDE EFFECTS:      None  
.: 245      0372 1 !--  
.: 246      0373 1 !  
.: 247      0374 1 !  
.: 248      0375 2 ! BEGIN  
.: 249      0376 2 ! LOCAL  
.: 250      0377 2 !     hold_khar,  
.: 251      0378 2 !     hold_lst_sp,  
.: 252      0379 2 !     hold_mra,  
.: 253      0380 2 !     hold_tsf,  
.: 254      0381 2 !     hold_lm,  
.: 255      0382 2 !     hold_wrd_ptr,  
.: 256      0383 2 !     ira_hold : VECTOR [10],  
.: 257      0384 2 !     hold_headers,  
.: 258      0385 2 !     minor_code,  
.: 259      0386 2 !     sca_hold :sca_definition;  
.: 260      0387 2 !  
.: 261      0388 2 ! Report pending errors.  
.: 262      0389 2 remneg ();  
.: 263      0390 2 remmrg ();  
.: 264      0391 2 tstblk (0);  
.: 265      0392 2 ! It's a chapter or appendix now.  
.: 266      0393 2 npagen [sct_typ] = (IF .counter_minor_type EQL min_chapt_inf  
.: 267      0394 3 ! THEN scf_chapt  
.: 268      0395 3 ! ELSE sct_append);  
.: 269      0396 2 !  
.: 270      0397 2 rskips (ira);           ! Skip spaces and tabs before the text.  
.: 271      0398 2 !  
.: 272      0399 2 ! We may have had chapters OR appendixes already if so begin  
.: 273      0400 2 ! counting where we left off.  
.: 274      0401 2 !  
.: 275      0402 2 !  
.: 276      0403 2 ! It is possible that we are in the no_page mode and this new chapter  
.: 277      0404 2 ! will throw a page. If the layout puts the page number at the bottom  
.: 278      0405 2 ! of the page then we can't update the NPAGEN until we are at the next  
.: 279      0406 2 ! page. Otherwise the page numbers come out incorrectly.  
.: 280      0407 2 ! For the utilities, we must keep the section numbering as it used to be.  
.: 281      0408 2 !  
.: 282      0409 2 npagen [sct_number] = .counter_value;           ! To maintain compatibility  
.: 283      0410 2 npagen [sct_page] = 1;                      ! number the next page "1"  
.: 284      0411 2 npagen [sct_sub_page] = 0;                  ! turn off subpaging.  
.: 285      0412 2 gca_chapt = true;                         ! Mark document as containing chapters/applications.  
.: 286      0413 2 !  
.: 287      0414 2 !  
.: 288      0415 2 ! Reset footnote number to 0.  
.: 289      0416 2 !  
.: 290      0417 2 fnct_number_l = 0;  
.: 291      0418 2 fnct_number_r = 0;  
.: 292      0419 2 fnct_number_ = 0;  
.: 293      0420 2 !  
.: 294      0421 2 ! See if this chapter is supposed to start on a new page.  
.: 295      U 0422 2 ! IF dsrplus !THEN  
.: 296      U 0423 2 ! IF NOT .new_page  
.: 297      U 0424 2 ! THEN  
.: 298      U 0425 2 ! BEGIN
```

```
: 299      U 0426 2           gtpc (.test_page_amount);
: 300      U 0427 2           outcrg (); !Now, force the paper to be positioned as it should.
: 301      U 0428 2           END
: 302      U 0429 2           ELSE
: 303      U 0430 2           BEGIN
: 304      U 0431 2           %FI
: 305      U 0432 2           !Initialization of new chapter or appendix
: 306      U 0433 2           %IF dsrplus %THEN
: 307      U 0434 2           IF .startodd
: 308      U 0435 2           THEN
: 309      U 0436 2           BEGIN
: 310      U 0437 2           fs_wchar (mra, rintes);
: 311      U 0438 2           fs_wchar (mra, %C'w');
: 312      U 0439 2           fs_wchar (mra, %C' ');
: 313      U 0440 2           tsf_int_vl = .tsf_int_vl + 3;
: 314      U 0441 2           END
: 315      U 0442 2           ELSE
: 316      U 0443 2           %FI
: 317      U 0444 2           gcpage (); !Start a new page
: 318      U 0445 2           %IF dsrplus %THEN
: 319      U 0446 2           !Turn off page headers for this page if not first title [always].
: 320      U 0447 2           IF NOT .hct_title_always
: 321      U 0448 2           THEN
: 322      U 0449 2           BEGIN
: 323      U 0450 2           %FI
: 324      U 0451 2           hold_headers = .hct_headers; !No header at top of chapters and appendices.
: 325      U 0452 2           hct_headers = false;
: 326      U 0453 2           %IF dsrplus %THEN
: 327      U 0454 2           END;
: 328      U 0455 2           %FI
: 329      U 0456 2           hct_odd_even = 0; !Reset odd/even page parity.
: 330      U 0457 2           %IF dsrplus %THEN
: 331      U 0458 2           END;
: 332      U 0459 2           %FI
: 333      U 0460 2           IF .sca_autotitle !If AUTOTITLE (default), then:
: 334      U 0461 2           THEN
: 335      U 0462 2           BEGIN !Clear subtitles
: 336      U 0463 3           LOCAL
: 337      U 0464 3           hold_tsf;
: 338      U 0465 3
: 339      U 0466 3
: 340      U 0467 3           hold_tsf = .tsf; !Remember current TSF.
: 341      U 0468 3           tsf = sbttsf; !Switch to the TSF for subtitles.
: 342      U 0469 3           tsf_int_hl = 0;
: 343      U 0470 3           tsf_ext_hl = 0;
: 344      U 0471 3           tsf = .hold_tsf; !Return to main TSF.
: 345      U 0472 3           fs_init (sbtmra); !Complete reset of subtitle MRA.
: 346      U 0473 3           END
: 347      U 0474 2           ELSE
: 348      U 0475 2           IF .gca_autosubt !Auto subtitle is on, don't carry
: 349      U 0476 2           THEN !over past subtitles.
: 350      U 0477 3           BEGIN !Clear subtitles
: 351      U 0478 3           LOCAL
: 352      U 0479 3           hold_tsf;
: 353      U 0480 3
: 354      U 0481 3           hold_tsf = .tsf; !Remember current TSF.
: 355      U 0482 3           tsf = sbttsf; !Switch to the TSF for subtitles.
```

```
: 356      0483 3      tsf_int_hl = 0;
: 357      0484 3      tsf_ext_hl = 0;
: 358      0485 3      tsf=.hold_tsf;
: 359      0486 3      fs init (sbfmra);
: 360      0487 2      END;
: 361      0488 2
: 362      0489 2      !Generate the specified number of lines before the header text.
: 363      0490 2      !This is only done if we are starting the chapter/append on a new page.
: 364      U 0491 2      %IF dsrplus %THEN
: 365      U 0492 2      IF .new_page
: 366      U 0493 2      THEN
: 367      U 0494 2      BEGIN
: 368      U 0495 2
: 369      U 0496 2      !Skip 9 lines after the title, 8 after the subtitle, else skip 12.
: 370      U 0497 2      IF .hct_title_always
: 371      U 0498 2      THEN
: 372      U 0499 2      IF .hct_subtitle
: 373      U 0500 2      THEN guskip (.lines_before - 4)
: 374      U 0501 2      ELSE guskip (.lines_before - 3)
: 375      U 0502 2
: 376      U 0503 2      ELSE guskip (.lines_before);
: 377      U 0504 2
: 378      U 0505 2      END
: 379      U 0506 2      ELSE
: 380      0507 2      guskip (.lines_before);           !end of top-of page-skip
: 381      0508 2
: 382      0509 2      !Reset header levels.
: 383      0510 2      INCR i FROM 1 TO .hllist [cl_max_index] DO
: 384      0511 2      hllist [.i] = 0;
: 385      0512 2
: 386      0513 2      hllist [cl_index] = 1;
: 387      0514 2
: 388      0515 2      !Reset entity counts for Example, Figure, Table.
: 389      0516 2      INCR i FROM 0 TO 2 DO
: 390      0517 2      ecc [.i, ecc$h_counter] = 0;
: 391      0518 2
: 392      0519 2      sdxy ();           !Reset some other things
: 393      0520 2
: 394      0521 2      IF .phan_top_first
: 395      0522 2      THEN
: 396      0523 2      !This was specified at the top of the very first page, and so
: 397      0524 2      !we cannot wait for NEWPAG to advance the page number.
: 398      0525 3      BEGIN
: 399      0526 3      pagen [sct_typ] = .npagen [sct_typ];
: 400      0527 3      pagen [sct_number] = .npagen [sct_number];
: 401      0528 3      pagen [sct_page] = .npagen [sct_page];
: 402      0529 3      npagen [sct_page] = .npagen [sct_page] + 1;
: 403      0530 2      END;
: 404      0531 2
: 405      0532 2      !If at the top of any page, output the page number. This is being done
: 406      0533 2      because of problems with the the design of RUNOFF. NEWPAG won't output
: 407      0534 2      the page number until this header text forces a title to be generated,
: 408      0535 2      which may be too late in this case.
: 409      0536 3      IF NOT (.gca_op_dev EQL op_dev_flip)
: 410      0537 2      THEN
: 411      0538 2      BEGIN
: 412      0539 3      IF .brn_open
```

```
: 413      0540 3      THEN
: 414          0541 3      IF .phan_top_first
: 415              0542 3      THEN
: 416                  0543 3      putppg (pagen, -1)
: 417                      0544 3      ELSE
: 418                          0545 3      putppg (npagen, -1);
: 419      END;
: 420
: 421      0547 2
: 422          0548 2      ! If creating a binary table of contents file, write out the counter and
: 423              0549 2      display descriptor.
: 424                  0550 2      IF .brn_open
: 425                      0551 2      THEN
: 426                          0553 2      !User is generating a table of contents.
: 427                              0554 2      putcnt ( .counter_major_type
: 428                                  ..counter_minor_type
: 429                                      ..tocpage
: 430                                          .0
: 431                                              ..counter_pre_string_length
: 432                                              ..counter_pre_string_ptr
: 433                                              ..counter_post_string_length
: 434                                              ..counter_post_string_ptr
: 435                      );
: 436
: 437          0563 2      !Save scanning information
: 438              0564 2      push_sca;    !Save the SAVED SCA bits.
: 439
: 440          0567 2      INCR i FROM 0 TO sca_size - 1 DO
: 441              0568 2          sca_hold [.i] = .sca [.i];
: 442                  0569 2          hold_lm = .sca_lm;           ! Save left margin; it might be used if
: 443
: 444          0571 2      INCR i FROM 0 TO 10 - 1 DO           centering and wrapping.
: 445              0572 2          ira_hold [.i] = .ira [.i];       ! Save IRA.
: 446                  0573 2          hold_khar = .khar;           ! Save KHAR.
: 447
: 448          0575 2      !+
: 449          0576 2      ! If either (1) this head will become a title or (2) user is creating a
: 450              0577 2      .BRN file, then we must scan the text twice. The first time we scan it at full
: 451                  0578 2      width, 150 characters, for title and/or TOC.
: 452                      0579 2      !!! If the title actually runs wider than the current right margin,
: 453                          0580 2      !!! however, we truncate it and append "...".
: 454
: 455          0582 2      IF .sca_autotitle OR .brn_open
: 456              0583 2      THEN
: 457                  0584 3      BEGIN
: 458
: 459          0586 3      sca_prescan = false;           !A ';' does NOT terminate this command.
: 460              0587 3      sca_rskips = true;            !Ignore multiple spaces and tabs.
: 461                  0588 3      sca_fill = false;
: 462                      0589 3      sca_justify = false;
: 463                          0590 3      !Set right margin big to catch long headers.
: 464                              0591 3      sca_lm = 0;
: 465                                  0592 3      sca_rm = 150;
: 466                                      0593 3      sca_do_ind = false;           !Ignore indexing for subtitle/TOC scan.
: 467
: 468          0595 3      !Make a title out of this chapter/appendix title, if that's what the
: 469              0596 3      user desires. The routine TITLES, which normally processes
```

```
; 470      0597 3      ! titles/subtitles, contains the necessary logic.
; 471      0598 3      IF .sca_autotitle
; 472      0599 3      THEN
; 473      0600 3      !Collect caption text in title buffer
; 474      0601 4      BEGIN
; 475      0602 4      !Set up bolding and underlining, if requested.
; 476      0603 5      IF (.caption_is_bold AND .sca_do_bld)
; 477      0604 4      THEN
; 478      0605 5      BEGIN
; 479      0606 5      sca_bld = true;
; 480      0607 5      sca_wrd_c_bld = true;
; 481      0608 4      END;
; 482      0609 4
; 483      0610 5      IF (.caption_is_underlined AND .sca_do_und)
; 484      0611 4      THEN
; 485      0612 5      BEGIN
; 486      0613 5      sca_und = true;
; 487      0614 5      sca_wrd_c_und = true;
; 488      0615 4      END;
; 489      0616 4
; 490      0617 4      setcas (.caption_case);      !Set up case rules for the caption text.
; 491      0618 4      titles (h_title);          !Use TITLES to get the title.
; 492      0619 3      END;
; 493      0620 3      !End of autotitle processing.
; 494      0621 3      IF .brn_open
; 495      0622 3      THEN
; 496      0623 4      BEGIN
; 497      0624 4
; 498      0625 4      !Restore IRA and KHAR.
; 499      0626 4      INCR i FROM 0 TO 10 - 1 DO
; 500      0627 4          ira [.i] = .ira_hold [.i];
; 501      0628 4
; 502      0629 4      INCR i FROM 0 TO sca_size - 1 DO
; 503      0630 4          sca [.i] = .sca_hold [.i];
; 504      0631 4
; 505      0632 4      pop_sca;      !Restore the SAVED SCA bits.
; 506      0633 4      !Save scanning information for another scan.
; 507      0634 4      push_sca;      !Save the SAVED SCA bits.
; 508      0635 4
; 509      0636 4      INCR i FROM 0 TO sca_size - 1 DO
; 510      0637 4          sca_hold [.i] = .sca [.i];
; 511      0638 4
; 512      0639 4      !Save IRA and KHAR.
; 513      0640 4      INCR i FROM 0 TO 10 - 1 DO
; 514      0641 4          ira_hold [.i] = .ira [.i];
; 515      0642 4
; 516      0643 4      !Going to TOC, may or may not be an auto-title.
; 517      0644 4      sca_prescan = false;      !A ';' does NOT terminate this command.
; 518      0645 4      sca_rskips = true;        !Ignore multiple spaces and tabs.
; 519      0646 4      sca_fill = false;
; 520      0647 4      sca_justify = false;
; 521      0648 4      !Set right margin big to catch long headers.
; 522      0649 4      sca_lm = 0;
; 523      0650 4      sca_rm = 150;
; 524      0651 4      sca_do_ind = false;      !Ignore indexing for title/TOC scan.
; 525      0652 4      sca_fc_case = true;
; 526      0653 4      sca_fc = true;
```

```
527      0654 4           khar = .hold_khar;
528      0655 4
529      0656 4           ! Switch to the Footnote TSF and MRA (which are not otherwise
530      0657 4           used in .ch/.ax) to collect the header information for the TOC.
531      0658 4
532      0659 4           hold_mra = .mra;
533      0660 4           mra = foomra;
534      0661 4           hold_tsf = .tsf;
535      0662 4           tsf = footsf;
536      0663 4
537      0664 4           !Before initializing the text descriptors, it is necessary to explicitly
538      0665 4           reset the maximum length of the footnote MRA. It is clobbered if any
539      0666 4           footnotes have been processed so far. See the comments in FNONLY around
540      0667 4           line 1545 for the detailed justification.
541      0668 4           fs_maxsize (mra) = s_fmra;
542      0669 4
543      0670 4           !Now initialize the text descriptors.
544      0671 4           fs init (mra);
545      0672 4           INCR i FROM 0 TO tsf_size - 1 DO
546      0673 4           tsf [.i] = 0;
547      0674 4
548      0675 4           !Set up bolding and underlining, if requested.
549      0676 5           IF (.caption_is_bold AND .sca_do_bld)
550      0677 4           THEN
551      0678 5           BEGIN
552      0679 5           sca_bld = true;
553      0680 5           sca_wrd_c_bld = true;
554      0681 4           END;
555      0682 4
556      0683 5           IF (.caption_is_underlined AND .sca_do_und)
557      0684 4           THEN
558      0685 5           BEGIN
559      0686 5           sca_und = true;
560      0687 5           sca_wrd_c_und = true;
561      0688 4           END;
562      0689 4
563      0690 4           !Set up case rules for the chapter/appendix.
564      0691 4           setcas (.caption_case);
565      0692 4
566      0693 4           !Scan 150-wide into footnote MRA.
567      0694 4           scant ();
568      0695 4           endwrd (false, false, false);
569      0696 4
570      0697 4           !Switch TSF and MRA back.
571      0698 4           mra = .hold_mra;
572      0699 4           tsf = .hold_tsf;
573      0700 3           END;
574      0701 3
575      0702 3           !Restore IRA and KHAR.
576      0703 3           INCR i FROM 0 TO 10 - 1 DO
577      0704 3           ira [.i] = .ira_hold [.i];
578      0705 3
579      0706 3           khar = .hold_khar;
580      0707 2           END;                                !End of title-or-TOC processing
581      0708 2
582      0709 2           INCR i FROM 0 TO sca_size - 1 DO
583      0710 2           sca [.i] = .sca_hold [.i];
```

```
584 0711 2
585 0712 2
586 0713 2      pop_sca; !Restore the SAVED SCA bits.
587 0714 2      sca_fill = true;           !Fill the header even if not filling text.
588 0715 2      sca_prescan = false;        !A ';' does NOT terminate this command.
589 0716 2      sca_rskips = true;          !Ignore multiple spaces and tabs.
590 0717 2
591 0718 3      !Set up bolding and underlining, if requested.
592 0719 2      IF (.caption_is_bold AND .sca_do_bld)
593 0720 3      THEN
594 0721 3          BEGIN
595 0722 3              sca_bld = true;
596 0723 2              sca_wrd_c_bld = true;
597 0724 2              END;
598 0725 3      IF (.caption_is_underlined AND .sca_do_und)
599 0726 2      THEN
600 0727 3          BEGIN
601 0728 3              sca_und = true;
602 0729 3              sca_wrd_c_und = true;
603 0730 2              END;
604 0731 2
605 0732 2      ! Generate the number for the .ch/.ax if not turned off.
606 0733 3      IF show_counter
607 0734 2      THEN
608 0735 2          !User didn't turn off the numbering, so generate the .ch/.ax counter.
609 0736 3          BEGIN
610 0737 3              !Generate the counter in using the case rules for the caption.
611 0738 3              outctr (.counter_minor_type, .caption_case);
612 0739 3
613 0740 3      !Put the counter into the output buffer.
614 0741 3      fs_next (fs01) = .fs_start ('s01);
615 0742 3
616 0743 3      INCR i FROM 1 TO .fs_length (fs01) DO
617 0744 4          BEGIN
618 0745 4              LOCAL
619 0746 4                  temp_char;
620 0747 4
621 0748 4                  fs_rchar (fs01, temp_char);
622 0749 4                  endchr (.temp_char);
623 0750 3                  END;
624 0751 3
625 0752 3      ! End the string so the spaces_after will not get underlined (if in
626 0753 3          ! effect).
627 0754 3          endwrd (false, false, false);
628 0755 3
629 0756 3      !Insert spaces after the counter if user didn't say BREAK or BETWEEN.
630 0757 3      !Do not put out more than 75 spaces regardless of what the user said.
631 0758 4      IF NOT (.break_before_caption OR (.lines_between GTR 0))
632 0759 3      THEN
633 0760 4          BEGIN
634 0761 4              LOCAL
635 0762 4                  sca_hold_c_bldun;
636 0763 4                  sca_hold_ac_bldun;
637 0764 4
638 0765 4      !ENDCHR seems to carry over underlining rules even for spaces.
639 0766 4      !We must turn off these bits before we add the extra spaces.
640 0767 4      sca_hold_c_bldun = .sca_wrd_c_bldun;
```

```
: 641      0768 4           sca_hold_ac_bln = .sca_wrd_ac_bln;  
: 642      0769 4  
: 643      0770 4           INCR i FROM 1 TO (MIN (.counter_spaces_after, 75)) DO  
: 644      0771 5           BEGIN  
: 645      0772 5           sca_wrd_c_bldun = 0;  
: 646      0773 5           sca_wrd_ac_bln = 0;  
: 647      0774 5           endchr (%C' );  
: 648      0775 4           END;  
: 649      0776 4  
: 650      0777 4           !Restore the SCA bits.  
: 651      0778 4           sca_wrd_c_bldun = .sca_hold_c_bldun;  
: 652      0779 4           sca_wrd_ac_bln = .sca_hold_ac_bln;  
: 653      0780 3           END;  
: 654      0781 3  
: 655      0782 3           !And now end this 'word'.  
: 656      0783 3           endwrd (false,false,false);  
: 657      0784 3  
: 658      0785 3           !If the user wants a break before the caption, we must  
: 659      0786 3           !make any adjustments if not flush_left.  
: 660      0787 3           IF .caption_is_centered OR .caption_is_flush_right  
: 661      0788 3           THEN  
: 662      0789 4           BEGIN  
: 663      0790 4  
: 664      0791 5           IF .break_before_caption OR (.lines_between GTR 0)  
: 665      0792 4           THEN  
: 666      0793 5           BEGIN  
: 667      0794 5           tsf_adjust =  
: 668      0795 6           ?IF .caption_is_centered !Center the counter.  
: 669      0796 6           THEN (.sca_rm = (.tsf_ext_hl))/2 !Ignore spaces_after  
: 670      0797 6           ELSE !Force the counter right.  
: 671      0798 5           .sca_rm = .tsf_ext_hl;  
: 672      0799 5           END  
: 673      0800 4           ELSE  
: 674      0801 5           BEGIN  
: 675      0802 5           !No break to be done so we must set the left margin so wrapping  
: 676      0803 5           !will left justify on the caption left margin.  
: 677      0804 5           hold_lm = .tsf_ext_hl; !Remember for use after SCA is restored.  
: 678      0805 5           sca_lm = .tsf_ext_hl;  
: 679      0806 4           END;  
: 680      0807 4           END  
: 681      0808 4  
: 682      0809 4  
: 683      0810 4           +  
: 684      0811 4           | The header is flush-left. If the caption is to immediately follow,  
: 685      0812 4           | set the left margin so that following text will wrap properly.  
: 686      0813 3  
: 687      0814 3           _  
: 688      0815 3           ELSE  
: 689      0816 4           !If we aren't doing a break after the counter, we must pull in  
: 690      0817 3           !the left margin to prevent writing over the counter.  
: 691      0818 3           IF NOT (.break_before_caption OR (.lines_between GTR 0))  
: 692      0819 3           THEN  
: 693      0820 3           !Set left margin beyond the counter so if the caption wraps,  
: 694      0821 3           !it will be nicely justified with itself.  
: 695      0822 3           sca_lm = .tsf_ext_hl;  
: 696      0823 2           END  
: 697      0824 2           !End of code to generate chapter counter.
```

```
: 698 0825 2      ! If we are not generating a number, then the first character
: 699 0826 2      | scanned should be the first character in this MRA.
: 700 0827 2
: 701 0828 2      sca_fc = true;
: 702 0829 2      !+
: 703 0830 2      | If the header is other than flush-left and user asked for break or
: 704 0831 2      | some lines between, we have already computed amount to shift it,
: 705 0832 2      | we now put it out on a line by itself.
: 706 0833 2      !-
: 707 0834 2      gca_line_pend = 1;      !Until we throw the header, a line is pending.
: 708 0835 2
: 709 0836 3      IF (.break_before_caption) OR (.lines_between GTR 0)
: 710 0837 2      THEN
: 711 0838 2          BEGIN
: 712 0839 3              outnj ();
: 713 0840 3          gcskip(.lines_between);      !Put out the counter.
: 714 0841 2          END;      !Skip the number of lines requested.
: 715 0842 2
: 716 0843 2      ! Write text to .BRN file. The call to PUTTXT is done here, between
: 717 0844 2      | generating the counter and the caption, so that coordination with
: 718 0845 2      | FLIP/BIND is maintained.
: 719 0846 2
: 720 0847 2      IF .brn_open
: 721 0848 2      THEN
: 722 0849 2          ! Write to .BRN (or .BFL) file.
: 723 0850 2
: 724 0851 2          putttx (.fs_length (foomra), .fs_start (foomra),
: 725 0852 2          .caption_major_type, .caption_minor_type);
: 726 0853 2
: 727 0854 2      !Set up case rules for the heading.
: 728 0855 2      setcas (.caption_case);
: 729 0856 2
: 730 0857 2      !Tell SCANT that the first character of the header is the first
: 731 0858 2      !character of a word. (ENDCHR turned this stuff off before.)
: 732 0859 2      sca_fc_case = true;
: 733 0860 2
: 734 0861 2      scant ();      !Go get the caption.
: 735 0862 2
: 736 0863 2      !SCA_WRD CPEND equals rintes IFF there was a trailing space/tab
: 737 0864 2      | after the text. In such a case ENDWRD has already been called.
: 738 0865 2      | Calling it again would have the effect of forcing an additional
: 739 0866 2      | space out into TSF/MRA.
: 740 0867 2      IF .sca_wrd_cpend NEQ rintes
: 741 0868 2      THEN
: 742 0869 2          endwrd (false, false, false)
: 743 0870 2
: 744 0871 2
: 745 0872 2      ELSE
: 746 0873 2
: 747 0874 2
: 748 0875 2      IF .sca_wrd_lst_und EQL 0
: 749 0876 2          AND
: 750 0877 2          .sca_wrd_lst_sp GTR 0
: 751 0878 2          THEN
: 752 0879 2          !Chop off trailing spaces/tabs. When doing so, also back up
: 753 0880 2          |intra-line pointer and counter appropriately. If justification
: 754 0881 3          |was in effect, also cancel the justification mark that got writ-  
ten onto the MRA.  
          !Note that trailing spaces that are underlined are not discarded.  
          BEGIN
```

```
: 755      0882 3      LOCAL
: 756      0883 3      chars_to_drop;
: 757      0884 3
: 758      0885 3      chars_to_drop = .sca_wrd_lst_sp +
: 759      0886 4          (-IF .sca_justify
: 760      0887 4              THEN 3
: 761      0888 4                  ELSE 0
: 762      0889 3              );
: 763      0890 3      fs_length (mra) = .fs_length (mra) - .chars_to_drop;
: 764      0891 3      fs_next (mra) = CH$PLUS (.fs_next (mra), -.chars_to_drop);
: 765      0892 3      sca_wrd_lst_sp = 0;
: 766      0893 2      END;
: 767      0894 2
: 768      0895 2      !Restore previous scanner rules and set standard rules.
: 769      0896 2      hold_wrd_ptr = .sca_wrd_ptr;           !Remember start of next word.
: 770      0897 2      hold_lst_sp = .sca_wrd_lst_sp;        !Remember last-space info.
: 771      0898 2
: 772      0899 2      INCR i FROM 0 TO sca_size - 1 DO
: 773      0900 2          sca [.i] = .sca_hold [.i];
: 774      0901 2
: 775      0902 2      pop_sca;           !Restore the SAVED SCA bits again.
: 776      0903 2      sca_wrd_ptr = .hold_wrd_ptr;       !Restore start of next word.
: 777      0904 2      sca_wrd_lst_sp = .hold_lst_sp;        !Restore last-space info.
: 778      0905 2
: 779      0906 2      !This section basically outputs the last line of the header. This
: 780      0907 2      may also be the first (if only one line long). What must happen here
: 781      0908 2      is if the header is flush-left, the last line (if different from the first,
: 782      0909 2      must start at the caption left margin. The same goes for centered
: 783      0910 2      captions if nobreak is in effect and if the caption is so long that
: 784      0911 2      it wraps. Now if it doesn't wrap, then this section will output
: 785      0912 2      the whole counter and caption. By using GCA_LINE_PEND, we may determine
: 786      0913 2      if this is the first line output. If a line has already gone out, then
: 787      0914 2      ! SCANT has already adjusted the left margin and no TSF_adjust is needed.
: 788      0915 2
: 789      0916 2      !If the header is other than flush-left, compute amount to shift it.
: 790      0917 2      !If user asked for a BREAK or had some LINES_BETWEEN, then the shift
: 791      0918 2      for the counter has already been taken care of and this section
: 792      0919 2      is only for the caption, otherwise the shift is for the whole header.
: 793      0920 2      IF .caption_is_centered OR .caption_is_flush_right
: 794      0921 2          THEN
: 795      0922 2
: 796      0923 2      IF .gca_line_pend                      !NO line thrown yet.
: 797      0924 2          THEN
: 798      0925 2              tsf_adjust =
: 799      0926 3                  (IF .caption_is_centered           !Center the counter/and/or caption.
: 800      0927 3                      THEN (.sca_rm - .tsf_ext_hl)/2
: 801      0928 3                      ELSE .sca_rm - .tsf_ext_hl )     !Force the counter/caption right.
: 802      0929 2          ELSE
: 803      0930 3              IF (.break_before_caption) OR (.lines_between GTR 0)
: 804      0931 2                  THEN
: 805      0932 2                      tsf_adjust =
: 806      0933 3                          (IF .caption_is_centered           !Center the counter/and/or caption.
: 807      0934 3                              THEN (.sca_rm - .tsf_ext_hl)/2
: 808      0935 2                              ELSE .sca_rm - .tsf_ext_hl);   !Force the counter/caption right.
: 809      0936 2
: 810      0937 2
: 811      0938 2      outnj ();                         !Force out the caption.
```

```
: 812      0939 2
: 813      0940 2      gcskip (.lines_after);
: 814      0941 2
: 815      0942 2      sca_sect_empty = true;           !This section is empty now.
: 816      0943 2
: 817      0944 2      !Set proper case conversion for remainder of section.
: 818      0945 2      setcas (.gca_case);
: 819      0946 2
: 820      0947 2      gca_pchax = false;          !Pending .NUMBER CHAPTER/APPENDIX done.
: 821      0948 2
: 822      U 0949 2      %IF dsrplus %THEN
: 823      U 0950 2      IF NOT .hct_title_always AND .new_page
: 824      U 0951 2      THEN
: 825      U 0952 2      %FI
: 826      U 0953 2      hct_headers = .hold_headers;      !Restore .HEADERS or .NO HEADERS status.
: 827      U 0954 2
: 828      U 0955 2      %IF dsrplus %THEN
: 829      U 0956 2      !We know now that the chapter counter has been output.
: 830      U 0957 2      !If we have moved to the top of a page fine, but if not,
: 831      U 0958 2      we must update the PAGEN structure. We update if we did not explicitly
: 832      U 0959 2      request a new page.
: 833      U 0960 2      IF NOT .new_page
: 834      U 0961 2      THEN
: 835      U 0962 2      BEGIN
: 836      U 0963 2      pagen [sct_number] = .counter_value;
: 837      U 0964 2      pagen [sct_typ] = (IF .counter_minor_type EQL min_chapt_inf
: 838      U 0965 2              THEN sct_chapt
: 839      U 0966 2              ELSE sct_append);
: 840      U 0967 2      npagen [sct_page] = .npagen [sct_page] + 1;
: 841      U 0968 2      END;
: 842      U 0969 2      %FI
: 843      0970 1      END;                      !End of OUTCHA
```

```
.TITLE OUTCHA Processor for chapter and appendix heade
      rs.
.IDENT \V04-000\
.PSECT $0WNS,NOEXE,2
00000 PP_SCA: .BLKB 48
.EXTRN RINTES, SFMRA, ECC
.EXTRN FNCT, F00MRA, FOOTSF
.EXTRN FS01, GCA, HCT, HLDSP
.EXTRN HLLIST, IRA, MRA
.EXTRN TITMRA, TITTSF, NPAGEN
.EXTRN PAGEN, PHAN, SBTMRA
.EXTRN SBTTSF, SCA, TSF
.EXTRN KWAR, ENDCHR, ENDWRD
.EXTRN GCPAGE, GCSKIP, GTPC
.EXTRN GUSKIP, OUTCRG, OUTCTR
.EXTRN OUTNJ, PACSEC, PACXXX
.EXTRN PUTCNF, PUTTPG, PUTTXT
.EXTRN REMMRG, REMNEG, RSKIPS
.EXTRN SCANT, SDXY, SETCAS
.EXTRN TITLE$, TSTBLK
```

								PSECT	SCODE\$, NOWRT, 2		
				OFFC	00000			.ENTRY	OUTCHA, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-	; 0313	
				5B	00000000G	EF	9E	00002	MOVAB	TSF, R11	
				5A	00000000	EF	9E	00009	MOVAB	PP SCA, R10	
				59	00000000G	EF	9F	00010	MOVAB	SCA+120, R9	
				SE	FE58	CE	9E	00017	MOVAB	-424(SP), SP	
		00000000G	EF	00	FB	0001C		CALLS	#0, REMNEG	0389	
		00000000G	EF	00	FB	00023		CALLS	#0, REMMRG	0390	
		00000000G	EF	7E	D4	0002A		CLRL	-(SP)	0391	
		00000000G	EF	10	14	01	FB	0002C	CALLS	#1, TSTBLK	
						05	D1	00033	CMPL	COUNTER_MINOR_TYPE, #16	
						05	D0	00037	BNEQ	1\$	
						03	D0	00039	MOVL	#1, R0	
						03	D0	0003E	BRB	2\$	
						50	F0	00041	MOVL	#3, R0	
	04			00	00000000G	EF	9F	0004A	INSV	R0, #0, #4, NPAGEN	
00000000G	EF	00000000G	EF	01	FB	00050		PUSHAB	IR4	0398	
		00000000G	EF	18	AC	D0	00057	CALLS	#1, RSKIP		
		00000000G	EF	01	D0	0005F		MOVL	COUNTER_VALUE, NPAGEN+4	0409	
		00000000G	EF	00000000G	EF	B4	00066	MOVL	#1, NPAGEN+8	0410	
		00000000G	EF	01	D0	0006C		CLRW	NPAGEN+2	0411	
		00000000G	EF	00000000G	EF	7C	00073	MOVL	#1, GCA+40	0412	
		00000000G	EF	00000000G	EF	D4	00079	CLRQ	FNCT+36	0417	
		00000000G	EF	58	00000000G	FF	D0	00086	FNCT+44	0419	
		00000000G	EF	00000000G	FF	D4	0008D	CALLS	#0, GCPAGE	0444	
		00000000G	EF	00000000G	EF	D4	00093	MOVL	@HCT+8, HOLD_HEADERS	0451	
		00000000G	EF	07	14	B9	E8	00099	CLRL	@HCT+8	0452
		00000000G	EF	2E	00000000G	FF	E9	0009D	BLBS	HCT+16	0456
		00000000G	EF	51	00000000G	6B	D0	000A4	BLBC	@SCA+140, 3\$	0461
		00000000G	EF	6B	00000000G	EF	9E	000A7	MOVL	@GCA+8, 4\$	0475
		00000000G	EF	50	00000000G	6B	D0	000AE	MOVAB	SBTTSF, TSF	0481
		00000000G	EF	6B	00000000G	60	7C	000B1	MOVL	TSF, R0	0482
		00000000G	EF	6B	00000000G	51	D0	000B3	CLRL	(R0)	0483
		00000000G	EF	00000000G	EF	D4	000B6	MOVL	HOLD_TSF, TSF	0485	
		00000000G	EF	00000000G	EF	9E	000BC	CLRL	SBTMRA+12	0486	
		00000000G	EF	04	04	AC	DD	000D2	MOVAB	SBTMRA+16, SBTMRA	
		00000000G	EF	51	00000000G	01	FB	000D5	PUSHL	SBTMRA, SBTMRA+4	
		00000000G	EF	51	00000000G	EF	D0	000DC	CALLS	LINES BEFORE	0507
		00000000G	EF	50	00000000G	50	D4	000E3	MOVL	#1, GOSKIP	
		00000000G	EF	07	11	11	000E5	CLRL	HLLIST, R1	0510	
		00000000G	EF	40	00000000G	04	D4	000F7	BRB	I	
	F5	00000000G	EF	50	00000000G	51	F3	000EE	5\$: CLRL	6\$	
		00000000G	EF	01	00000000G	01	D0	000F2	AOBLEQ	HLLIST+4[1]	0511
		00000000G	EF	50	00000000G	50	D4	000F9	MOVL	R1, I, 5\$	0513
		00000000G	EF	24	24	C5	000FB	5\$: CLRL	#1, HLLIST+4	0516	
		00000000G	EF	00	00000000G	00	FB	000FF	MULL3	I	0517
		00000000G	EF	02	02	F3	00109	INSV	#36, I, R1		
		00000000G	EF	00	00000000G	00	FB	0010D	AOBLEQ	#0, #24, #32, ECC+12[R1]	
		00000000G	EF	1E	00000000G	EF	E9	00114	CALLS	#2, I, 7\$	
		00000000G	EF	00	00000000G	EF	F0	0011B	BLBC	#0, SDXY	0519
	04	00000000G	EF	00	00000000G	EF	7D	00128	INSV	PHAN+24, 8\$	0521
		00000000G	EF	00	00000000G	EF	7D	00128	MOVO	NPAGEN, #0, #4, PAGEN	0526
		00000000G	EF	00	00000000G	EF	7D	00128		NPAGEN+4, PAGE#4	0527

OUTCHA
V04-000Processor for chapter and appendix headers.
OUTCHA -- Process chapter & appendix headersG 9
16-Sep-1984 01:19:03
14-Sep-1984 13:07:31
VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI;1Page 19
(4)OUTI
V04

02	00000000G	EF	00000000G	04	EF	D6	00133		INCL	NPAGEN+8	: 0529			
					04	ED	00139	8\$:	CMPZV	#4, #4, GCA+208, #2	: 0536			
					26	13	00142		BEQL	11\$				
				3E	5C	AC	E9	00144	BLBC	BRN_OPEN, 12\$: 0539			
				0B	00000000G	EF	E9	00148	BLBC	PHAR+24, 9\$: 0541			
				7E	00000000G	01	CE	0014F	MNEG	#1, -(SP)	: 0543			
					00000000G	EF	9F	00152	PUSHAB	PAGEN				
						09	11	00158	BRB	10\$				
						01	CE	0015A	9\$:	MNEG	#1, -(SP)			
						02	FB	00163	10\$:	PUSHAB	NPAGEN			
				00000000G	EF	9F	0015D		CALLS	#2, PUTTPG				
					18	5C	AC	E9	0016A	BLBC	BRN_OPEN, 12\$			
					7E	2C	AC	7D	0016E	MOVQ	COUNTER_POST_STRING_LENGTH, -(SP)			
					7E	24	AC	7D	00172	MOVQ	COUNTER_PRE_STRING_LENGTH, -(SP)			
						7E	D4	00176	CLRL	-(SP)	: 0554			
						70	AC	DD	00178	PUSHL	TOCPAGE			
						10	AC	7D	0017B	MOVQ	COUNTER_MAJOR_TYPE, -(SP)			
						08	FB	0017F	CALLS	#8, PUTCNT	: 0554			
						6A	EC	B9	DO 00186	12\$:	MOVL	@SCA+100, PP_SCA		
						04	AA	F0	B9	DO 0018A	MOVL	@SCA+104, PP_SCA+4		
						08	AA	F4	B9	DO 0018F	MOVL	@SCA+108, PP_SCA+8		
						0C	AA	F8	B9	DO 00194	MOVL	@SCA+112, PP_SCA+12		
						10	AA	FC	B9	DO 00199	MOVL	@SCA+116, PP_SCA+16		
						14	AA	00	B9	DO 0019E	MOVL	@SCA+120, PP_SCA+20		
						18	AA	04	B9	DO 001A3	MOVL	@SCA+124, PP_SCA+24		
						1C	AA	08	B9	DO 001A8	MOVL	@SCA+128, PP_SCA+28		
						20	AA	0C	B9	DO 001AD	MOVL	@SCA+132, PP_SCA+32		
						24	AA	10	B9	DO 001B2	MOVL	@SCA+136, PP_SCA+36		
						28	AA	14	B9	DO 001B7	MOVL	@SCA+140, PP_SCA+40		
						2C	AA	18	B9	DO 001BC	MOVL	@SCA+144, PP_SCA+44		
								50	D4	001C1	CLRL	I		
								88	A940	DO 001C3	13\$:	SCA[I], SCA_HOLD[I]		
F2						6E40	50	0000005F	8F	F3	001C9	AOBLEQ	#95, I, 13\$	
						57	FC	B9	DO 001D1		MOVL	@SCA+116, HOLD_LM		
								50	D4	001D5	CLRL	I		
F2						D8	AD40	00000000G	EF	D0	001D7	14\$:	MOVL	IRA-4[I], IRA_HOLD[I]
						50	09	F3	001E1		AOBLEQ	#9, I, 14\$		
						54	00000000G	EF	DO	001E5		MOVL	KHAR, HOLD_KHAR	
						07	14	B9	E8	001EC		BLBS	@SCA+140, T5\$	
						03	5C	AC	E8	001FO		BLBS	BRN_OPEN, 15\$	
							01D6	31	001F4		BRW	29\$		
						54	A9	34	A9	D4	001F7	15\$:	CLRL	SCA+172
								01	DO	001FA		MOVL	#1, SCA+204	
								F0	B9	D4	001FE		CLRL	@SCA+104
								EC	B9	D4	00201		CLRL	@SCA+100
								FC	B9	D4	00204		CLRL	@SCA+116
						00	B9	96	8F	9A	00207		MOVZBL	#150, @SCA+120
						30	A9	08	8A	0020C		BICB2	#8, SCA+168	
						36	14	B9	E9	00210		BLBC	@SCA+140, 18\$	
						0C	4C	AC	E9	00214		BLBC	CAPTION_IS_BOLD, 16\$	
						08	30	A9	E9	00218		BLBC	SCA+168, 18\$	
						20	A9	01	88	0021C		BISB2	#1, SCA+152	
						4C	A9	01	88	00220		BISB2	#1, SCA+196	
08						30	OD	50	AC	E9	00224	16\$:	BLBC	CAPTION_IS_UNDERLINED, 17\$
								01	E1	00228		BBC	#1, SCA+168, 17\$	
						20	A9	02	88	0022D		BISB2	#2, SCA+152	
						4C	A9	02	88	00231		BISB2	#2, SCA+196	

OUTCHA
V04-000

Processor for chapter and appendix headers.
OUTCHA -- Process chapter & appendix headers

H 9
16-Sep-1984 01:19:0
14-Sep-1984 13:07:3

VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI;1

Page 20
(4)

OUT
V04

00000000G	EF	3C	AC	DD	00235	17\$:	PUSHL	CAPTION CASE							0617
	7E	CC	01	FB	00238		CAILS	#1 SETCAS							0618
00000000G	EF	01	8F	9A	0023F		MOVZBL	#204 -(SP)							
	03	5C	AC	E8	00243	18\$:	CALLS	#1, TITLES							0621
		0165	31	0024E			BLBS	BRN_OPEN, 19\$							
F2	00000000GEF40	50	D8	AD40	DO	00253	19\$:	BRW	27\$						0626
			09	F3	0025D	20\$:	CLRL	I						0627	
			50	D4	00261		MOVL	IRA_HOLD[I], IRA-4[I]						0629	
F2	88 A940	50	0000005F	6E40	DO	00263	21\$:	AOBLEQ	#9, I, 20\$						0630
	EC	B9	8F	F3	00269		CLRL	I							
			6A	DO	00271		MOVL	SCA_HOLD[I], SCA[I]							
	FO	B9	04	AA	DO	00275	MOVL	#95, I, 21\$							
	F4	B9	08	AA	DO	0027A	MOVL	PP_SCA, @SCA+100							
	F8	B9	0C	AA	DO	0027F	MOVL	PP_SCA+4, @SCA+104							
	FC	B9	10	AA	DO	00284	MOVL	PP_SCA+8, @SCA+108							
	00	B9	14	AA	DO	00289	MOVL	PP_SCA+12, @SCA+112							
	04	B9	18	AA	DO	0028E	MOVL	PP_SCA+16, @SCA+116							
	08	B9	1C	AA	DO	00293	MOVL	PP_SCA+20, @SCA+120							
	0C	B9	20	AA	DO	00298	MOVL	PP_SCA+24, @SCA+124							
	10	B9	24	AA	DO	0029D	MOVL	PP_SCA+28, @SCA+128							
	14	B9	28	AA	DO	002A2	MOVL	PP_SCA+32, @SCA+132							
	18	B9	2C	AA	DO	002A7	MOVL	PP_SCA+36, @SCA+136							
	6A	EC	B9	DO	002AC		MOVL	PP_SCA+40, @SCA+140							
	04	AA	F0	B9	DO	002B0	MOVL	PP_SCA+44, @SCA+144							
	08	AA	F4	B9	DO	002B5	MOVL	@SCA+100, PP_SCA							
	0C	AA	F8	B9	DO	002BA	MOVL	@SCA+104, PP_SCA+4							
	10	AA	FC	B9	DO	002BF	MOVL	@SCA+108, PP_SCA+8							
	14	AA	00	B9	DO	002C4	MOVL	@SCA+112, PP_SCA+12							
	18	AA	04	B9	DO	002C9	MOVL	@SCA+116, PP_SCA+16							
	1C	AA	08	B9	DO	002CE	MOVL	@SCA+120, PP_SCA+20							
	20	AA	0C	B9	DO	002D3	MOVL	@SCA+124, PP_SCA+24							
	24	AA	10	B9	DO	002D8	MOVL	@SCA+128, PP_SCA+28							
	28	AA	14	B9	DO	002DD	MOVL	@SCA+132, PP_SCA+32							
	2C	AA	18	B9	DO	002F2	MOVL	@SCA+136, PP_SCA+36							
			50	D4	002F7		MOVL	@SCA+140, PP_SCA+40							
							MOVL	@SCA+144, PP_SCA+44							
							CLRL	I							
F2	6E40	50	0000005F	88 A940	DO	002E9	22\$:	MOVL	SCA[I], SCA_HOLD[I]						0636
							AOBLEQ	#95, I, 22\$						0637	
							CLRL	I							
F2	D8 AD40	50	00000000GEF40	DO	002F9	23\$:	MOVL	IRA-4[I], IRA_HOLD[I]							0640
							AOBLEQ	#9, I, 23\$						0641	
							CLRL	I							
	54	A9	34	A9	D4	00307	MOVL	SCA+172							0644
							CLRL	#1, SCA+204						0645	
							MOVL	@SCA+104						0646	
							CLRL	@SCA+100						0647	
							CLRL	@SCA+116						0648	
	00	B9	96	8F	9A	00317	MOVZBL	#150, @SCA+120						0650	
	30	A9	08	8A	0031C		B1CB2	#8, SCA+168						0651	
	58	A9	01	DO	00320		MOVL	#1, SCA+208						0652	
	1C	A9	01	DO	00324		MOVL	#1, SCA+148						0653	
00000000G	EF		54	DO	00328		MOVL	HOLD_KHAR, KHAR						0654	
U00000000G	53	00000000G	EF	DO	0032F		MOVL	MRA, HOLD MRA						0659	
	EF	00000000G	EF	9E	00336		MOVAB	FOOMRA, MRA						0660	
	52		6B	DO	00341		MOVL	TSF, HOLD TSF						0661	
	6B	00000000G	EF	9E	00344		MOVAB	FOOFSF, TSF						0662	
	50	00000000G	EF	DO	0034B		MOVL	MRA, R0						0668	

	08	A0	00000000G	8F	D0	00352	MOVL	#S FMRA, 8(R0)		
				0C	A0	D4 0035A	CLRL	12(R0)	0671	
	04	60	10	A0	9E	0035D	MOVAB	16(R0), (R0)		
		A0		60	D0	00361	MOVL	(R0), 4(R0)		
				50	D4	00365	CLRL	I		
F8		50	00 BB40	D4	00367	24\$:	CLRL	@TSF[I]		
		0C		27	F3	00368	A0BLEQ	#39 I, 24\$		
		08		4C	AC	E9 0036F	BLBC	CAPTION_IS BOLD, 25\$	0676	
		20	A9	30	A9	E9 00373	BLBC	SCA+168, 25\$		
		4C	A9	01	88	00377	BISB2	#1, SCA+152	0679	
	08	OD		4C	A9	01 88 0037B	BISB2	#1, SCA+196	0680	
		30	A9	50	AC	E9 0037F	BLBC	CAPTION IS UNDERLINED, 26\$	0683	
		20	A9	01	E1	0C383	BBC	#1, SCA+168, 26\$		
		4C	A9	02	88	00388	BISB2	#2, SCA+152	0686	
				02	88	0038C	BISB2	#2, SCA+196	0687	
				3C	AC	DD 00390	PUSHL	CAPTION CASE	0691	
		0000000CG	EF		01	FB 00393	CALLS	#1, SETCAS		
		00000000G	EF		00	FB 0039A	CALLS	#0, SCANT	0694	
					7E	7C 003A1	CLRQ	-(SP)	0695	
		00000000G	EF		7E	D4 003A3	CLRL	-(SP)		
		00000000G	EF		03	FB 003A5	CALLS	#3, ENDWRD	0698	
		00000000G	EF		53	D0 003AC	MOVL	HOLD_MRA, MRA		
				6B	52	D0 003B3	MOVL	HOLD_TSF, TSF	0699	
F2		00000000GEF40		50	D4	003B6	27\$:	CLRL	I	
				09	F3	003C2	MOVL	IRA_HOLD[I], IRA-4[I]	0703	
		00000000G	EF		54	D0 003C6	A0BLEQ	#9, I, 28\$		
					50	D4 003CD	MOVL	HOLD_KHAR, KHAR	0706	
	F2	88 A940		6E40	D0	003CF	28\$:	CLRL	I	0709
					30\$:	MOVL	SCA_HOLD[I], SCA[I]			
				50	8F	F3 003D5	A0BLEQ	#95, I, 30\$		
				EC	89	6A D0 003D9	MOVL	PP_SCA, @SCA+100		
				F0	89	04 AA D0 003E1	MOVL	PP_SCA+4, @SCA+104		
				F4	89	08 AA D0 003E6	MOVL	PP_SCA+8, @SCA+108		
				F8	89	0C AA D0 003EB	MOVL	PP_SCA+12, @SCA+112		
				FC	89	10 AA D0 003F0	MOVL	PP_SCA+16, @SCA+116		
				00	89	14 AA D0 003F5	MOVL	PP_SCA+20, @SCA+120		
				04	89	18 AA D0 003FA	MOVL	PP_SCA+24, @SCA+124		
				08	89	1C AA D0 003FF	MOVL	PP_SCA+28, @SCA+128		
				0C	89	20 AA D0 00404	MOVL	PP_SCA+32, @SCA+132		
				10	89	24 AA D0 00409	MOVL	PP_SCA+36, @SCA+136		
				14	89	28 AA D0 0040E	MOVL	PP_SCA+40, @SCA+140		
				18	89	2C AA D0 00413	MOVL	PP_SCA+44, @SCA+144		
				F0	B9	01 D0 00418	MOVL	#1, @SCA+104		
					34	A9 D4 0041C	CLRL	SCA+172	0713	
				54	A9	01 D0 0041F	MOVL	#1, SCA+204	0714	
					0C	4C E9 00423	BLBC	CAPTION_IS BOLD, 31\$	0715	
					08	30 A9 E9 00427	BLBC	SCA+168, 31\$	0716	
					20	A9 01 88 0042B	BISB2	#1, SCA+152	0721	
					4C	A9 01 88 0042F	BISB2	#1, SCA+196	0722	
	08	30	A9		50	AC E9 00433	31\$:	BLBC	CAPTION IS UNDERLINED, 32\$	0723
		20	A9			01 E1 00437	BBC	#1, SCA+168, 32\$		
		4C	A9			02 88 0043C	BISB2	#2, SCA+152	0728	
		00	38	AC		02 88 00440	BISB2	#2, SCA+196	0729	
7E		50				01 7A 00444	EMUL	#1, CAPTION_MINOR_TYPE, #0, -(SP)	0733	
						03 7B 0044A	EDIV	#3, (SP)+, R0, R0		
						02 50 D1 0044F	CMPL	R0 #2		
						0F 13 00452	BEQL	33\$		

**OUTCHA
V04-000**

Processor for chapter and appendix headers.
OUICHA -- Process chapter & appendix headers

J 9
16-Sep-1984 01:19:03 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 13:07:31 [RUNOFF.SRC]OUT[CHA.BLI;1]

Page 22
(4)

OUT
VOL

			03	38	AC	D1	00454		CMPL	CAPTION_MINOR_TYPE, #3	
			04	38	AC	D1	0045A		BEQL	33\$	
					03	13	0045E		CMPL	CAPTION_MINOR_TYPE, #4	
				00F0	31	00460		BEQL	33\$		
				3C	AC	DD	00463	33\$:	BRW	48\$	
				14	AC	DD	00466		PUSHL	CAPTION_CASE	
			00000000G	EF	02	FB	00469		PUSHL	COUNTER_MINOR_TYPE	0738
			00000000G	EF	00	DO	00470		CALLS	#2 OUTCTR	
			00000000G	53	EF	DO	00478		MOVL	FS01, FS01+4	0741
					52	D4	00482		MOVL	FS01+12, R3	0743
					1C	11	00484		CLRL	I	
				50	FF	9A	00486	34\$:	BRB	35\$	
					00000000G	EF	D6	0048D	MOVZBL	@FS01+4, TEMP_CHAR	0748
					00000000G	EF	D7	00493	INCL	FS01+4	
					00000000G	EF	50	DD	DECL	FS01+12	
			E0	00000000G	EF	01	FB	0049B	PUSHL	TEMP_CHAR	0749
					52	53	F3	004A2	CALLS	#1, ENDCHR	
						7E	7C	004A6	AOBLEQ	R3, I, 34\$	0743
						7E	D4	004A8	CLRQ	-(SP)	0754
				00000000G	EF	03	FB	004AA	CLRL	-(SP)	
						54	D4	004B1	CALLS	#3, ENDWRD	
						64	AC	D5	TSTL	R4	0758
						02	15	004B3	BLEQ	LINES_BETWEEN	
						54	D6	004B8	INCL	36\$	
						54	C8	004BA	BISL2	R4	
						42	E8	004BE	BLBS	BREAK BEFORE_CAPTION, R4	
						02	00	EF	EXTZV	R4, 40\$	
						02	00	EF	EXTZV	#0, #2, SCA+196, SCA_HOLD_C_BLDUN	0767
				0000004B	52	20	AC	DO	MOVL	#0, #2, SCA+200, SCA_HOLD_AC_BLUN	0768
					8F	52	D1	004D1	CMPL	COUNTER_SPACES_AFTER, R2	0770
						04	15	004D8	BLEQ	R2, #75	
						52	48	8F	MOVZBL	37\$, #75, R2	
						11	11	004E0	CLRL	I	
						4C	A9	03	BRB	39\$	
						50	A9	8A	BICB2	#3, SCA+196	0772
							03	8A	BICB2	#3, SCA+200	0773
				00000000G	EF	20	DD	004E6	PUSHL	#32	0774
					EB	53	01	FB	CALLS	#1, ENDCHR	
					50	00	F3	004EC	AOBLEQ	R2, I, 38\$	
					02	00	56	F0	INSV	SCA_HOLD_C_BLDUN, #0, #2, SCA+196	0770
						55	F0	004F7	INSV	SCA_HOLD_AC_BLUN, #0, #2, SCA+200	0778
				00000000G	EF	7E	7C	00503	CLRQ	-(SP)	0779
						7E	D4	00505	CLRL	-(SP)	0783
						03	FB	00507	CALLS	#3, ENDWRD	
						04	AC	E8	BLBS	CAPTION_IS_CENTERED, 41\$	0787
						30	44	E9	BLBC	CAPTION_IS_FLUSH_RIGHT, 46\$	
						50	6B	DO	MOVL	TSF, R0	0796
						05	60	E8	BLBS	BREAK_BEFORE_CAPTION, 42\$	
						64	AC	D5	TSTL	LINE_BETWEEN	0791
						1E	15	00519	BLEQ	45\$	
						51	6B	DO	MOVL	TSF, R1	0793
						08	40	E9	BLBC	CAPTION_IS_CENTERED, 43\$	0796
						50	A0	C3	SUBL3	4(R0), ASCA+120, R0	
							04	C3	SUBL3	#2 R0	
							06	11	DIVL2	44\$	
								05	BRB	4(R0), ASCA+120, R0	0798
								06	SUBL3		

OUTCHA
V04-000

Processor for chapter and appendix headers.
OUTCHA -- Process chapter & appendix headers

K 9
16-Sep-1984 01:19
14-Sep-1984 13:07

VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI:1

Page 23
(4)

OUT
V04

28	A1	50	D0 0053A	44\$:	MOVL	R0 40(R1)	0795
	57	04	A0 00540	45\$:	BRB	49\$	0791
		06	11 00544		MOVL	4(R0), HOLD_LM	0804
	0E	54	E8 00546	46\$:	BRB	47\$	0805
	50	68	D0 00549		BLBS	R4, 49\$	0816
FC	B9	04	A0 0054C	47\$:	MOVL	TSF, R0	0820
		04	11 00551		MOVL	4(R0), @SCA+116	0733
1C	A9	01	D0 00553	48\$:	BRB	49\$	0828
00000000G	EF	01	D0 00557	49\$:	MOVL	#1, SCA+148	0834
	05	60	AC E8 0055E		MOVL	#1, GCA+224	0836
		64	AC D5 00562		BLBS	BREAK BEFORE CAPTION, 50\$	
			11 15 00565		TSTL	LINES_BETWEEN	
00000000G	EF	00	FB 00567	50\$:	BLEQ	51\$	
00000000G	EF	64	AC DD 0056E		CALLS	#0, OUTNJ	0839
00000000G	EF	01	FB 00571		PUSHL	LINES_BETWEEN	0840
	17	5C	AC E9 00578	51\$:	CALLS	#1, GCSKIP	0847
	7E	34	AC 7D 0057C		BLBC	BRN OPEN, 52\$	0852
		00000000G	EF DD 00580		MOVQ	CAPTION_MAJOR_TYPE, -(SP)	0851
		00000000G	EF DD 00586		PUSHL	FOOMRA	
00000000G	EF	04	FB 0058C		PUSHL	FOOMRA+12	
		3C	AC DD 00593	52\$:	CALLS	#4, PUTTXT	
00000000G	EF	01	FB 00596		PUSHL	CAPTION_CASE	0855
	58	A9	01 D0 0059D		CALLS	#1, SETCAS	0859
00000000G	EF	00	FB 005A1		MOVL	#1, SCA+208	0861
00000000G	8F	00A0	C9 D1 005A8		CALLS	#0, SCANT	0867
			0D 13 005B1		CMPL	SCA+280, #RINTES	
			7E 7C 005B3		BEQL	53\$	
			7E D4 005B5		CLRQ	-(SP)	0869
00000000G	EF	03	FB 005B7		CLRL	-(SP)	
		30	11 005BE		CALLS	#3, ENDWRD	
		00DC	C9 D5 005C0	53\$:	BRB	56\$	
			2A 12 005C4		TSTL	SCA+340	0872
		00D4	C9 D5 005C6		BNEQ	56\$	
			24 15 005CA		TSTL	SCA+332	0874
	05	EC	B9 E9 005CC		BLEQ	56\$	
	50		03 D0 005D0		BLBC	@SCA+100, 54\$	0886
			02 11 005D3		MOVL	#3, R0	
			50 D4 005D5	54\$:	BRB	55\$	
				55\$:	CLRL	R0	
51		50	00D4 00000000G	C9 C1 005D7	ADDL3	SCA+332, R0, CHARS_TO_DROP	
	OC	A0	EF D0 005DD		MOVL	MRA, R0	0890
	04	A0	51 C2 005E4		SUBL2	CHARS_TO_DROP, 12(R0)	
			51 C2 005E8		SUBL2	CHARS_TO_DROP, 4(R0)	0891
		00D4	C9 D4 005EC		CLRL	GCA+332	0892
	51	0080	C9 D0 005F0	56\$:	MOVL	SCA+248, HOLD_WRD_PNTR	0896
	52	00D4	C9 D0 005F5		MOVL	SCA+332, HOLD_LST_SP	0897
			50 D4 005FA		CLRL	I	0899
F2	88	A940	6E40 D0 005FC	57\$:	MOVL	SCA_HOLD[I], SCA[I]	0900
		50 0000005F	8F F3 00602		AOBLEQ	#95, I, 57\$	
	EC	B9	6A D0 0060A		MOVL	PP_SCA, @SCA+100	
	F0	B9	04 AA D0 0060E		MOVL	PP_SCA+4, @SCA+104	
	F4	B9	08 AA D0 00613		MOVL	PP_SCA+8, @SCA+108	
	F8	B9	0C AA D0 00618		MOVL	PP_SCA+12, @SCA+112	
	FC	B9	10 AA D0 0061D		MOVL	PP_SCA+16, @SCA+116	
	00	B9	14 AA D0 00622		MOVL	PP_SCA+20, @SCA+120	
	04	B9	18 AA D0 00627		MOVL	PP_SCA+24, @SCA+124	
	08	B9	1C AA D0 0062C		MOVL	PP_SCA+28, @SCA+128	

OUTCHA
VO4-000Processor for chapter and appendix headers.
OUTCHA -- Process chapter & appendix headersL 9
16-Sep-1984 01:19:03
14-Sep-1984 13:07:31VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI;1Page 24
(4)OUT
VO4

0C	89	20	AA	DO	00631	MOVL	PP-SCA+32, @SCA+132	
10	89	24	AA	DO	00636	MOVL	PP-SCA+36, @SCA+136	
14	89	28	AA	DO	0063B	MOVL	PP-SCA+40, @SCA+140	
18	89	2C	AA	DO	00640	MOVL	PP-SCA+44, @SCA+144	
0080	C9	51	DO	00645	MOVL	HOLD_WRD_PNTR, SCA+248		
00D4	C9	52	DO	0064A	MOVL	HOLD_LST_SP, SCA+332		
04	40	AC	E8	0064F	BLBS	CAPTION_IS_CENTERED, 58\$		
2C	44	AC	E9	00653	BLBC	CAPTION_IS_FLUSH_RIGHT, 62\$		
09	00000000G	EF	E8	00657	58\$:	BLBS	GCA+224, 59\$	
05	60	AC	E8	0065E	BLBS	BREAK_BEFORE_CAPTION, 59\$		
	64	AC	D5	00662	TSTL	LINES_BETWEEN		
		1C	15	00665	BLEQ	62\$		
	50	6B	DO	00667	59\$:	MOVL	TSF, R0	
51	00	08	40	AC	E9 0066A	BLBC	CAPTION_IS_CENTERED, 60\$	
	89	04	A0	C3	0C66E	SUBL3	4(R0), @SCA+120, R1	
	51	02	C6	00674		DIVL2	#2, R1	
51	00	B9	04	A0	C3 00679	60\$:	SUBL3	4(R0), @SCA+120, R1
	28	A0	51	DO	0067F	61\$:	MOVL	R1, 40(R0)
	00000000G	EF	00	FB	00683	62\$:	CALLS	#0, OUTNJ
			08	AC	DD 0068A		PUSHL	LINES_AFTER
	00000000G	EF	01	FB	0068D		CALLS	#1, GESKIP
	3C	A9	01	DO	00694		MOVL	#1, SCA+180
	00000000G	EF	00000000G	FF	DD 00698		PUSHL	@GCA+128
	00000000G	EF	00000000G	01	FB 0069E		CALLS	#1, SETCAS
	00000000G	FF	00000000G	EF	D4 006A5		CLRL	GCA+60
				58	DO 006AB		MOVL	HOLD_HEADERS, @HCT+8
				04	006B2		RET	

: Routine Size: 1715 bytes. Routine Base: \$CODE\$ + 0000

```
: 844      0971 1
: 845      0972 1 END
: 846      0973 0 ELUDOM
```

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$OUNS	48	NOVEC, WRT, RD : NOEXE,NOSHR, LCL, REL: CON,NOPIC,ALIGN(2)
\$CODE\$	1715	NOVEC,NOWRT, RD : EXE,NOSHR, LCL, REL: CON,NOPIC,ALIGN(2)

Library Statistics

File	----- Symbols -----			Pages Mapped	Processing Time
	Total	Loaded	Percent		
\$_255\$DUA28:[SYSLIB]XPORT.L32:1	590	0	0	252	00:00.1

OUTCHA
VO4-000 Processor for chapter and appendix headers.
OUTCHA -- Process chapter & appendix headers M 9
16-Sep-1984 01:19:03
14-Sep-1984 13:07:31 VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]OUTCHA.BLI;1
:_\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1 1248 137 10 86 00:00.3

Page 25
(4)

OUT
VO4

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:OUTCHA/OBJ=OBJ\$:OUTCHA MSRC\$:OUTCHA/UPDATE=(ENH\$:OUTCHA)

Size: 1715 code + 48 data bytes
Run Time: 00:39.9
Elapsed Time: 01:14.9
Lines/CPU Min: 1462
Lexemes/CPU-Min: 17712
Memory Used: 378 pages
Compilation Complete

0346 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

